

# PODSTAWY PROGRAMOWANIA W PYTHON

Dzień 13



# AGENDA

## DAY 13

- Return vs. print
- praca z plikami
- Dostarczanie argumentów do programu z konsoli
- webscrapping

# testowanie kodu

return vs print

# return vs print

- Funkcje z returnem - zwracają wartości - stąd możemy łatwo sprawdzić, czy podany input prawidłowo został obliczony i zwrócony output
- Funkcje z printem – na podany input odpowiadają drukowaniem informacji na ekranie, jak przetestować czy output jest prawidłowy?

Trzeba przechwycić output, zanim trafi na ekran

# | praca z plikami

# OS

- moduł os służy do pracy z plikami, ścieżkami, zmiennymi systemowymi

mkdir, chdir, getcwd, unlink, rmdir, listdir, walk

- os.path – działania na ścieżkach  
split, join, abspath

<https://docs.python.org/3/library/os.html>

<https://docs.python.org/3/library/os.path.html>

# shutil

wrapper dla poleceń systemowych, w pewnych sytuacjach ułatwia wykonanie poleceń

copytree, move, rmtree,

<https://docs.python.org/3/library/shutil.html>

# send2trash

przenosi do kosza

```
send2trash.send2trash(plik)
```



# argumenty programów

```
python plik.py  
python plik.py opcja1 opcja2 opcja3
```

```
import sys
```

**sys.argv** – lista argumentów przekazanych do Python

```
['plik.py', 'opcja1', 'opcja2', 'opcja3']
```

**sys.argv[0]** – ścieżka uruchomionego pliku

**sys.argv[1:]** – lista samych opcji

# | webscrapping

# HTML, CSS. JS

HTML – znacznikowy język tworzący strukturę stron internetowych

CSS – arkusz styli – odpowiada za wygląd strony

JavaScript (nie mylić z Java) – język kliencki (wykonywany przez przeglądarkę) – odpowiada za animację, wykonywanie skryptów

<https://www.w3schools.com/>

# Developer tools

## F12

Narzędzia do inspekcji stron internetowych  
F12

# requests, beautiful soup 4

- beautiful soup 4

pakiet do parsowania i wyszukiwania elementów  
html/xml

- requests

moduł do wysyłania żądań do serwerów www  
`response = requests.get('trojmiasto.pl')`

# bs4

<code>soup.select('#author')</code>	element z id = author
<code>soup.select('.notice')</code>	element o klasie = notice
<code>soup.select('div span')</code>	elementy <span> wewnątrz elementu <div>
<code>soup.select('div &gt; span')</code>	elementy <span> bezpośrednio wewnątrz elementu <div>, żaden inny element nie może wystąpić pomiędzy
<code>soup.select('input[name]')</code>	elementy <input>, które mają atrybut name z dowolną wartością
<code>soup.select('input[type="buton"]')</code>	elementy <input>, które mają atrybut type z wartością "buton"

`soup.select('p #author')`

# AUTOMATYZACJA ZADAŃ

- Windows – Task Scheduler (harmonogram zadań)

Aplikacja okienkowa w Panelu sterowania

Z linii komend – polecenie **schtasks**

[Link1](#)

[Link2](#)

- Mac / Linux – Crontab

Linia poleceń

[link1](#)

[link2](#)



# Thanks!!